National Defense
Defence nationale

UNCLASSIFIED

DREV REPORT 4318/84
FILE: 3633A-017
JANUARY 1984

CRDV RAPPORT 4318/84
DOSSIER: 3633A-017
JANVIER 1984

A QUAD TREE DERIVED INDICATOR FOR INFRARED BACKGROUND ANALYSIS AND CLASSIFICATION

R. Charpentier

Centre de Recherches pour la Defense
Defence Research Establishment
Valcartier Quebec

BUREAU - RECHERCHE ET DEVELOPPEMENT
MINISTERE DE LA DEFENSE NATIONALE
CANADA

RESEARCH AND DEVELOPMENT BRANCH
DEPARTMENT OF NATIONAL DEFENCE
CANADA

NON CLASSIFIÉ

Canada

DREV R-4318/84
FILE: 3633A-017

UNCLASSIFIED

CRDV R-4318/84
DOSSIER: 3633A-017

A QUAD TREE DERIVED INDICATOR FOR INFRARED BACKGROUND ANALYSIS AND CLASSIFICATION

by

R. Charpentier

CENTRE DE RECHERCHES POUR LA DEFENSE

DEFENCE RESEARCH ESTABLISHMENT

VALCARTIER

Tel: (418) 844-4271

Québec, Canada

January/janvier 1984

NON CLASSIFIE

ABSTRACT

A new geometric indicator is proposed for analyzing and classifying infrared images. The indicator, which was named 'formity', provides an objective estimate of the 'nonuniformity' of an image in a very compact form. Derived from the quad tree theory, this indicator is relatively simple to apply and should be particularly useful in relation to large banks of images. Some typical results obtained from thermal images are presented to illustrate its properties.

RESUME

Un nouvel indicateur géométrique a été conçu et mis au point pour permettre l'analyse et la classification de grandes banques d'images infrarouges. Il permet d'estimer la 'non-uniformité' dans une image de façon objective et relativement simple et d'en exprimer le résultat sous forme très compacte. Il a été nommé indicateur de 'formité' et est dérivé de la théorie de la segmentation de type "quad tree". Quelques résultats obtenus avec des images infrarouges sont présentés pour illustrer les propriétés de cet indicateur.

TABLE OF CONTENTS

ABSTRACT/RESUME i

1.0 INTRODUCTION 1

2.0 QUAD TREE TYPE SEGMENTATION 2

2.1 Intuitive Approach 2
2.2 Our Algorithm 8
2.3 Threshold Calculation 15

3.0 APPLICATIONS AND DISCUSSION 16

3.1 Origin of the Data 16
3.2 Typical Results and Discussion 17
3.3 The Total Formity Indicator 22

4.0 CONCLUSION 25

5.0 ACKNOWLEDGEMENTS 25

6.0 REFERENCES 26

FIGURES 1 to 7

TABLES I to IV

APPENDIX A - Decomposition of Total Formity Indicators 27

1.0 INTRODUCTION

The power spectral density (PSD), also known as the Weiner spectrum, is generally used for analyzing geometric structures in infrared (IR) backgrounds (BG). It is a well-known analysis and is commercially available from most image processing equipment manufacturers.

However, for an extensive data base, the PSD analysis has several disadvantages. Its calculation requires a large-memory computer as well as many floating point operations, making its use very costly in terms of time and equipment. Furthermore, the amount of output data is very large, and the final interpretation is generally subjective as it is done by a programmer-analyst.

For our studies of IR backgrounds, we decided to develop an alternative to the PSD. The geometric indicator that we propose has three main objectives:

a) simplicity; i.e. limited number of floating point calculations,
b) hardware oriented; i.e. uses small computer memory, and
c) objectivity; i.e. small amount of output data requiring no human interpretation.

We call this new indicator, which is derived from the quad tree theory, the 'formity' indicator. It is described in this report first by using an intuitive approach (Section 2.1), and then a more systematic and powerful approach (Section 2.2). Some results are discussed in Chapter 3.0.

This work was performed at DREV between January and March 1983 under PCN 33A17, Teal Ruby and HI Camp Experiments.

2.0 QUAD TREE TYPE SEGMENTATION

2.1 Intuitive Approach

2.1.1 Introduction

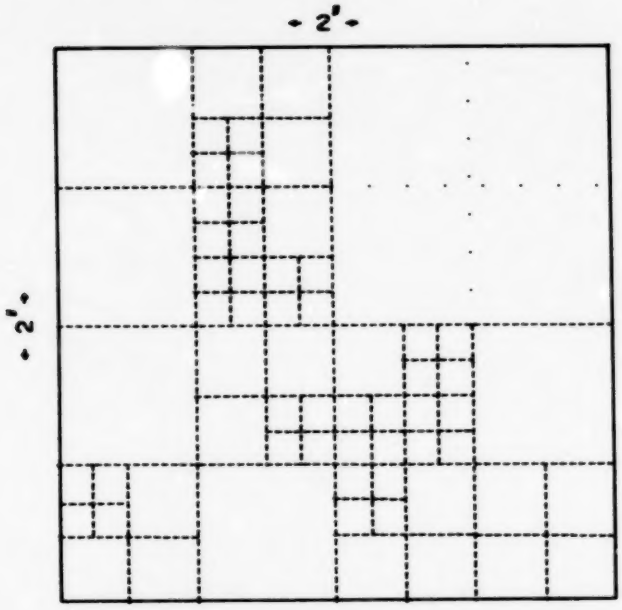
Assume an image of 2^n rows, each one having 2^n pixels (i.e. 2^n x 2^n array). The quad tree is an approach to image representation based on the successive subdivision of an image into quadrants (Fig. 1a). To calculate the quad tree of an image, we repeatedly subdivide the array into quadrants, subquadrants, etc. until we obtain homogeneous blocks which consist of a group of pixels having a relatively uniform gray level. The theoretical limit of the segmentation process is the single pixel.

This process is represented in Fig. 1b by a tree of out-degree 4 whose root node stands for the entire array, the four sons of the root node represent the quadrants, and the terminal nodes correspond to the blocks of the array where no further subdivision is necessary.

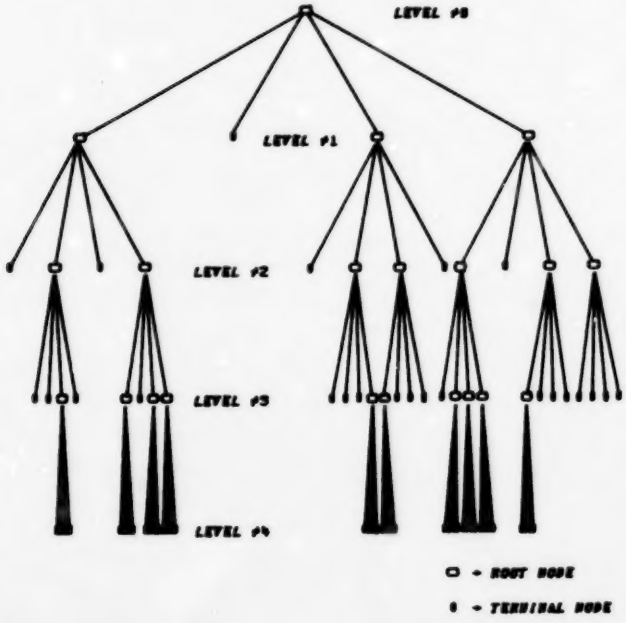
Strictly speaking, the quad tree is defined in a unique form for binary imagery where the terminal nodes contain only '0' pixels or '1' pixel. For multiple gray level images, the concept of a variance threshold is required, as shown in Subsection 2.1.2.

2.1.2 Quad Tree Example

Figure 1a shows a simple example of the recursive segmentation process. At the beginning, a vertical picture line and a horizontal one are taken in the middle of the image. The variance of the pixels of the two data lines is calculated and then compared with a



a) Segmented image



b) Corresponding quad tree representation

FIGURE 1 - Quad tree type segmentation

predetermined threshold. When the actual variance exceeds the threshold, image is segmented into four subimages. Subsequently for each subimage, two lines are taken in a similar way so that the variance threshold test can be applied.

In Fig. 1a, we see no further segmentation in the upper right quadrant indicating that the local variance of this subimage is lower than the threshold. For all other quadrants, the segmentation process goes on as long as required by the variance threshold test.

2.1.3 Origin and Present Uses

The quad tree was first introduced to compress binary imagery by replacing a large uniform subimage by its node characteristics. Thus, the quad tree representation of an image is generally more compact than the conventional raster representation.

Numerous applications of the quad tree, such as data compression and approximation (Ref. 1) and the representation of geographical information (Ref. 2), are reported in the literature.

For our studies, the quad tree is used to derive a nonuniformity indicator described in the next section.

2.1.4 Nonuniformity Indicator

It is interesting to observe that the quad tree representation of an image is not of constant length. The length depends on the image complexity estimated using the variance threshold test.

Figure 1 gives an example of a segmented image. Let us define level 0 as the whole image level, level 1 as the first segmentation level (i.e. four subimages), level 2 as the second segmentation level (i.e. 16 subimages), and so on.

For each level, we know the maximum number of possible subimages and we can count the number of actual subimages as given by the quad tree calculation (i.e. intermediate nodes). Dividing these two numbers for each level we get an indicator of complexity in the image.

For simplicity, the prefix 'nonuni' was eliminated from 'nonuniformity' and thus this indicator is called 'formity'. From now on in the text, the term 'formity' is used for 'nonuniformity' and corresponds to exactly the same concept.

Table I gives the results obtained using the image in Fig. 1.

Plotting the formity indicator (Fig. 2), we observe a monotonically decreasing curve whose general shape is determined by the degree of formity in the image.

An alternative way of representing the formity curve is to count the number of terminal nodes at each level (i.e. the number of subimages in which there is no further segmentation) as shown in Table II and Fig. 3. It is easy to prove that a simple integration relation exists between the two methods of calculating the formity. In this latter case, we see the composition of the image for each level of segmentation. It is interesting to observe that each level can be associated with a '2-D spatial frequency' (doubling with each segmentation). A correlation between the discrete Fourier transform component and the normalized level formity seems conceivable and will be studied in future work.

TABLE I

Results obtained using Figure 1

Intermediate node method

Level	Maximum subimages possible	Actual subimages	Integrated normalized formity
0	1	1	1.
1	4	4	1.
2	16	12	0.75
3	64	28	0.4375
4	256	40	0.15625
.			
.			
.			
	4^i		4^i
i	4	α	$\alpha/4$

TABLE II

Results obtained using Figure 1

Terminal node method

Level	Maximum subimages possible	Actual subimages with no further segmentation	Derived normalized formity
0	1	0	0.0
1	4	1	0.25
2	16	5	0.3125
3	64	18	0.28125
4	256	40	0.15625
:			
.			
	4^i		4^i
i	4	β	$\beta/4$

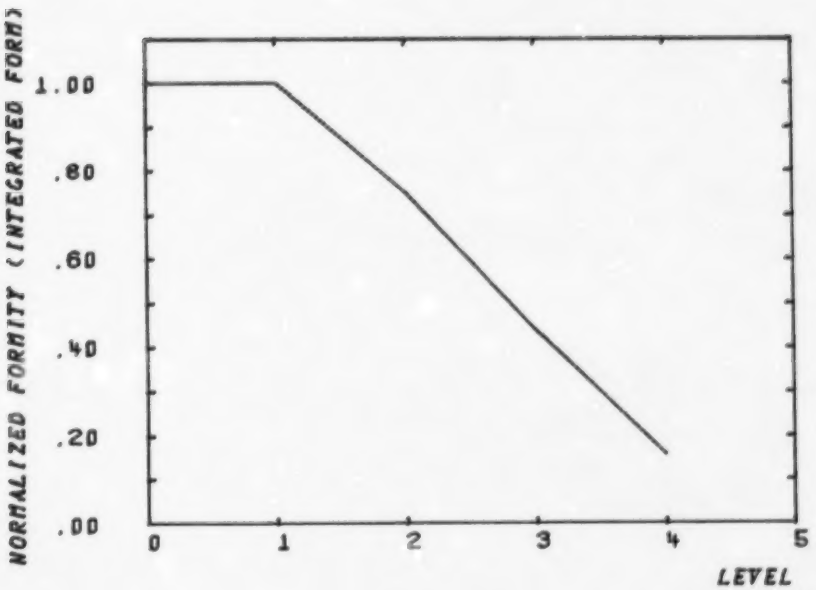


FIGURE 2 - Formity of each level (i.e. intermediate node method)

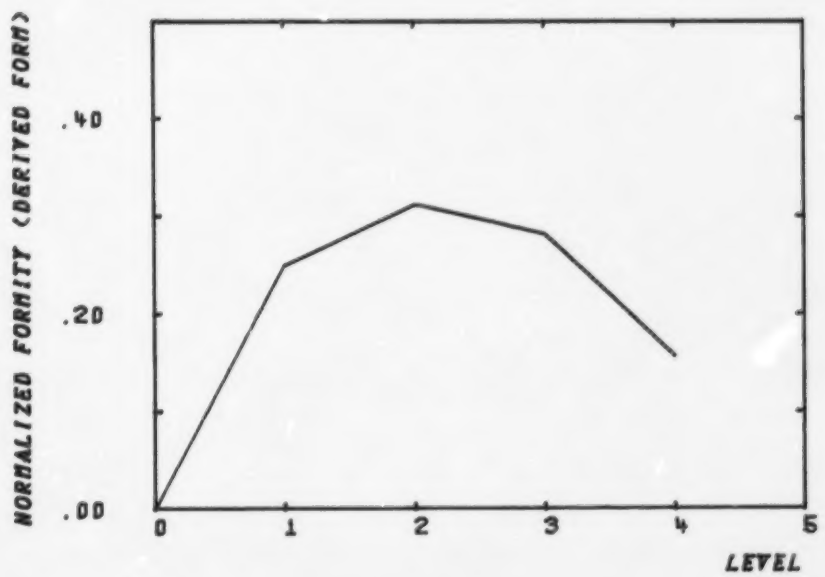


FIGURE 3 - Formity of each level (i.e. terminal node method)

Results obtained from real IR imagery are presented and discussed in Section 3.2.

2.2 Our Algorithm

2.2.1 Disadvantages of the Intuitive Approach

It is apparent that the intuitive approach for calculating the quad tree is not optimal. First, some structure can be missed by the variance threshold test since only two lines (a vertical and a horizontal one) of the image are used. Second, the segmentation process requires that the whole image be retained in memory, which occupies a large amount of space. Third, the processing time depends on image complexity.

In order to avoid these difficulties, a different approach to quad tree calculation was developed, and it is described in the next section.

2.2.2 Recursive Merging Process

The basic idea of the recursive merging process is to assume from the beginning that a complete quad tree exists for all levels. As the data come in, the mean and the variance are calculated for groups of four pixels. The variance threshold test is then applied within each group, and the formity counter is incremented, if necessary, as shown in Subsection 2.2.4. The mean and the variance of all groups are then transmitted to the next highest level.

Figure 4 shows on a very simple 16 x 16 pixel image how the recursive merging process works.

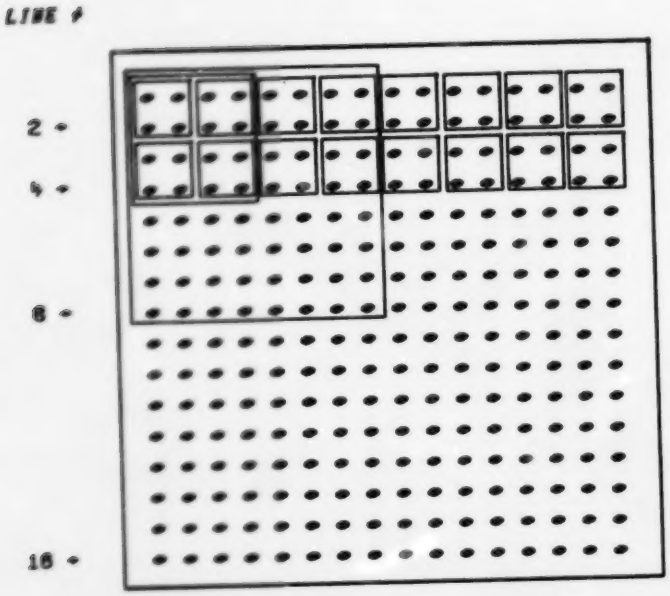


FIGURE 4 - Recursive merging for quad tree calculation

For a 16 x 16 pixel image, only four levels are possible since the pixel level is reached at level #4.

In each pair of lines (i.e. two consecutive serial lines), the mean and the variance can be calculated for eight groups of four pixels, and the threshold test can be applied to calculate the formity at level #3. After treating the fourth line, it is possible to merge four groups of pixels, which allows a formity calculation at level #2. After the eighth line, a formity calculation is done at level #1. Finally, after the sixteenth line, the formity can be calculated for all levels of the image including level #0.

A more detailed description of the recursive merging algorithm and the formity calculation process is given in Subsection 2.2.4.

2.2.3 Advantages of Recursive Merging

The recursive merging process for calculating the quad tree has numerous advantages over the intuitive approach. The most important ones are:

a) All pixels are "seen"; no structure can be missed.

b) The procedure is systematic; the same calculations are used at each level and they consist in merging four groups of data into a larger one. Thus it is possible to imagine some sort of hardware accelerator that would do this merging calculation very efficiently.

c) The recursive merging leads to the exact mean and variance of the data. These two results are extremely important for infrared background analysis and for the performance evaluation of image processing.

d) The processing time is independent of the complexity of the image. In most cases, it is shorter than that required with the intuitive approach since many calculations can be done in parallel within the same level as well as for two independent levels.

e) The algorithm is serial; a limited memory is required. For a typical 256 x 256 pixel image, the total buffer memory required is only 1K instead of 128K for the PSD calculation.

f) Everything is done in one pass of the data. For real-time operation, this feature is very important.

2.2.4 Recursive Merging Algorithm

Let us consider an image of 256 x 256 pixels as our standard image size. As our algorithm is serial, a large memory of 64K is not required to store the image, but the following buffers are used:

BUF8(2,256)	at level #8 (i.e. 2 image lines)
BUF7(128,2)	#7 (i.e. mean + variance)
BUF6(64,2)	#6 (i.e. mean + variance)
BUF5(32,2)	#5 (i.e. mean + variance)
BUF4(16,2)	#4 (i.e. mean + variance)
BUF3(8,2)	#3 (i.e. mean + variance)
BUF2(4,2)	#2 (i.e. mean + variance)
BUF1(2,2)	#1 (i.e. mean + variance)
BUF0(1,2)	#0 (i.e. global mean and global variance)

Thus, a total memory of only 1K is required. The following flags are used:

F7	at level #7
F6	#6
.	.
.	.
.	.
F1	#1
F0	#0

These indicate the number of updates done at each level (i.e. $0 < F^* < 4$).

Finally, the variable FORM(I) where I = 1, 2,...,8 is used to store the number of subimage nodes at each level (i.e. intermediate node method). This formity counter will be used to calculate the formity indicator.

With these buffers and variables, the serial quad tree algorithm for the calculation of the formity is:

```
BEGINNING:  I ← 0                          ; Reset of the image counter
NEW IMAGE: I ← I + 1                       ; Reset of flags
    BUF*(*,*) ← 0                          ; Reset of buffers
    FORM(*) ← 0                            ; Reset of formity counter
    L ← 0                                  ; Reset of line counter
READ:  BUF8 ← DATA                         ; Two image lines
    L ← L + 2
    TEST #8                                ; Variance threshold test
    FORM(8) ◁                              ; Update of the formity ind.
    BUF7 ◁ BUF8                            ; Update at level #7
    IF L/2 IS ODD, GO TO READ
    TEST #7
    FORM(7) ◁                              ; Update of the formity ind.
    BUF6 ◁ BUF7                            ; Update at level #6

    IF L/4 IS ODD, GO TO READ
    ...                                    ; Test and updates at #5
    IF L/8 IS ODD, GO TO READ
    ...                                    ; Test and updates at #4
    IF L/16 IS ODD, GO TO READ
    ...                                    ; Test and updates at #3
    IF L/32 IS ODD, GO TO READ
    ...                                    ; Test and updates at #2
    IF L/64 IS ODD, GO TO READ
    ...                                    ; Test and updates at #1
    IF L/128 IS ODD, GO TO READ
    ...                                    ; Test and updates at #0
    PRINTER ← FORM(*)
    ...                                    ; Finalization of current image.
    GO TO NEW IMAGE
```

The test and update procedure is basically the same for each level. It may be described in more detail using the update at level #5 as an example:

```
Now BUF6 contains fresh results
    BUF5 is cleared
    F5 = 0

Previous level ...
    BUF5 ◁ BUF6                    ; 2 blocks of fresh data are
                                     merged in BUF5
    F5 ← F5 + 2

    If F5 = 2, GO TO READ          ; Same test as 'L/8 is odd'

    IF BUF5(*,2) < SF, GO TO CONT5 ; The variance threshold test is
                                     performed for all groups when
                                     BUF5 is complete.

    FORM(6) = FORM(6) + 4          ; The formity indicator is
                                     incremented for all groups
                                     requiring it.

CONT5:  CONTINUE
NEXT LEVEL: ...
```

2.2.5 Description of the Recursive Merging Algorithm

At the beginning, all counters and buffers are cleared to prepare for the arrival of the first of two lines of data in BUF8. The local mean and the local variance are calculated for each group of four

pixels as shown in Subsection 2.2.2 so that the variance threshold test can be made in each group. The formity counter at level #8 is then incremented accordingly, and the local mean and variance are transmitted as partial results to BUF7.

If BUF7 is not complete for testing and updating of the formity counter at level #7, two new lines are read and treated in the same way. When BUF7 is complete, the merging of four groups of four pixels is possible. The test and update procedure shown in the previous section is very similar for all levels.

For all other levels, the test and update procedure is performed as soon as the buffer of the active level is complete. Note, however, that there are two alternative ways of deciding if a buffer is complete. The first one is especially interesting for hardware implementation, but the second one is more practical for software development.

The first one is based on the line counter. Since the number of lines in each level is a power of two, the completeness of an active buffer is found by dividing the line counter by $2^{(8-i)}$ where i is the active level number. The main algorithm shown in Subsection 2.2.4 illustrates how the test is performed. In a hardware implementation, one simple logical gate on the adequate bit of the line counter (binary counter) could be used to control access to the corresponding level in a very simple and efficient manner.

The second possible test uses an auxiliary flag counter (i.e. F8, F7,...,F0) and is incremented each time some new results enter the active level. The procedure for test and update, shown in the previous section, illustrates how the test is performed.

2.3 Threshold Calculation

It is evident that the formity indicator calculation relies mainly on the variance threshold test. Thus, the choice of the appropriate threshold is very important and will greatly influence the results. As a result, a threshold calculation procedure is proposed that is directly applicable to the infrared imagery analysis.

In order to avoid segmentation due to image noise, it was decided to fix the variance threshold higher than the variance due to noise.

Let n be the number of elements for which the variance is calculated,

NEGL the Noise Equivalent Temperature Difference expressed in gray levels, and

K a constant factor fixed by the analyst to avoid sensitivity to noise.

Then all pixels can be expressed as fluctuations around the mean as follows:

$$X(I) = MEAN \pm (K \times NEGL) \qquad [1]$$

The corresponding variance threshold is given by

$$VT = \frac{\sum^{n} (X(I)-MEAN)^2}{n-1} \qquad \text{(variance definition)}$$

using [1]

$$= \frac{\sum^{n} (\text{MEAN} \pm (K \times \text{NEGL}) - \text{MEAN})^2}{n - 1}$$

$$VT = \frac{\sum^{n} (\pm K \times \text{NEGL})^2}{n - 1}$$

$$= \frac{\sum^{n} K^2 \times \text{NEGL}^2}{n - 1}$$

$$= \frac{n}{n - 1} K^2 \ \text{NEGL}^2 \qquad [2]$$

Equation 2 permits the calculation of the variance threshold for each level. The factor K can be arbitrarily fixed to any value chosen by the analyst. In our studies, we chose K = 2.33 by interpreting [1] as the tolerance limit equation used in engineering. With Gaussian variation around the mean, the 2.33 factor corresponds to a 98% level of confidence as shown by the integration of the normal distribution function. Reference 3 provides further discussions about the confidence limit calculation and refers to other good books on statistics applied to engineering problems.

It is very important to understand that, for all formity calculations, the variance threshold must be specified along with the results to insure their interpretability.

3.0 APPLICATIONS AND DISCUSSION

3.1 Origin of the Data

The data used in this chapter were obtained from the Canadian Centre of Remote Sensing. A Daedalus infrared line scanner, installed

in a Falcon aircraft, generates the thermal imagery which is digitized and transferred onto computer-compatible tape.

Table III summarizes the important features of the system. A standard format of 256 x 256 pixels has been chosen for program development and preliminary analysis.

TABLE III

Daedalus 1260 multispectral line scanner

Spatial resolution:	2.5 mrad IFOV
Field of view:	73.7° with S-bend correction
Roll compensation:	± 5°
Calibration:	two temperature controlled blackbodies
Scan speed:	selectable 12.5, 25, 50, 100 scans per second
Airborne data recording:	14-track Mincom tape and transferred to 9-track computer-compatible tape
Detectors:	InSb (LN2 cooled) 4.5-5 μm HgCdTe (LN2 cooled) 9-13 μm

3.2 Typical Results and Discussion

A large number of images were processed in order to obtain a variety of backgrounds (and, of course, of formity results). Two typical cases are presented and discussed in this section. The first one corresponds to a high formity case, and the second to a low formity case.

Figure 5 shows the normalized formity of each level for a high formity image using four different variance thresholds (i.e. VT = 2, 8, 128, 1024). To obtain these curves, the number of quad tree nodes of each level was divided by the maximum possible number of nodes of the level leading to the so-called normalized formity. Because all nodes are counted for each level, this has been named the intermediate node method.

The variance threshold of height (i.e. VT = 8) is the one suggested in Section 2.3 to avoid segmentation due to electro-optical noise. The thresholds of 2, 128 and 1024 are also used to show the dependence of the quad tree calculation on the threshold choice.

The curves obtained have a shape very similar to that obtained with the PSD calculation since they are monotonically decreasing functions of the level, and hence of the spatial frequency (cf Subsection 2.1.4).

For the low formity image (Fig. 6), the curves start to fall much earlier, indicating that the lower levels are not totally segmented.

Comparing the low formity image with the high formity one, some very interesting observations can be highlighted.

Firstly, the global variance of the low formity image is almost three times higher than that of the high formity one. This is caused by the occurrence of two large uniform patches in the low formity image. The formity remains low at levels #8 to #4 since the local variance within those patches is low, but important increases at levels #3 to #0 occur when the two patches are merged by the quad tree calculation process. The final variance of the low formity image is then fairly high even if the formity remains low. This indicates that the global variance of an image and its formity are independent of each

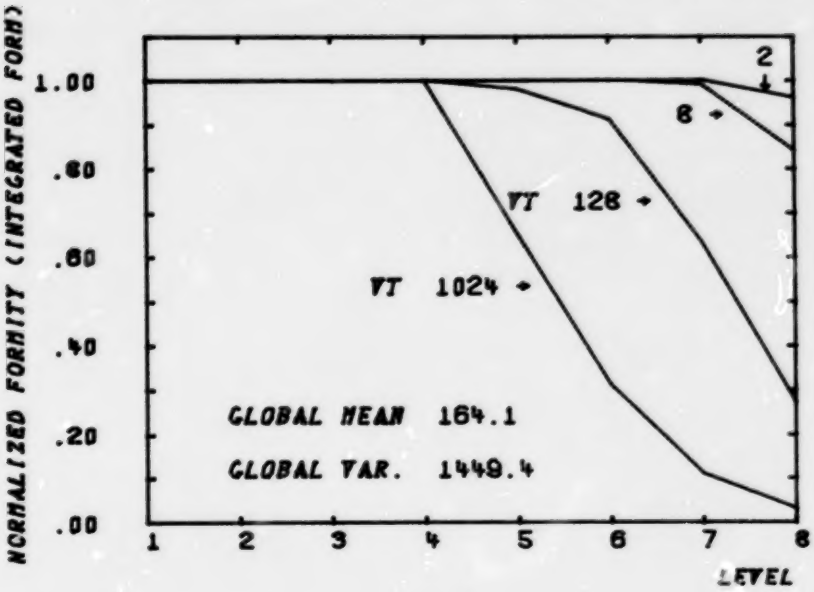


FIGURE 5 - Typical high formity image (intermediate node method)

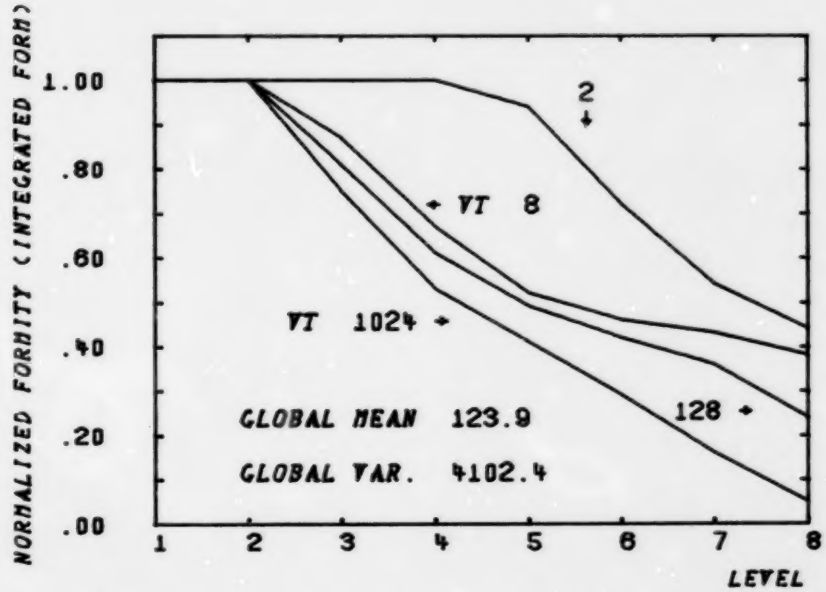


FIGURE 6 - Typical low formity image (intermediate node method)

other. The global variance is a radiometric indicator and the formity, a geometric one.

The second interesting observation concerns the threshold. In Fig. 6 we see that the curves for the thresholds of 8, 128, and 1024 have a similar behaviour, but the curve corresponding to the threshold of two (VT = 2) is significantly different. This is a result of the electro-optical noise in the image since the recommended variance threshold (Section 2.3) for avoiding noise effects is eight (VT = 8). The considerable separation between the curve with VT = 2 and the others (Fig. 6) indicates the importance of choosing a variance threshold higher than the variance due to noise especially in low formity imagery, where large patches have a fairly uniform gray level.

This effect is even more obvious in Fig. 7 where the formity has been calculated using the terminal node method. With this method the formity is obtained by dividing the number of 'uniform' subimages containing no further segmentation (i.e. terminal nodes) by the maximum number of possible images within each level (Subsection 2.1.4).

We clearly see in Fig. 7 that the formity peak for VT = 2 is shifted to levels #5 and #6, which leads to a misevaluation of the real image formity situated mainly at levels #2, #3 and #4.

Many other interesting observations were made on other images and will be discussed in future publications. In this report, the observations brought up are sufficient to illustrate the formity properties.

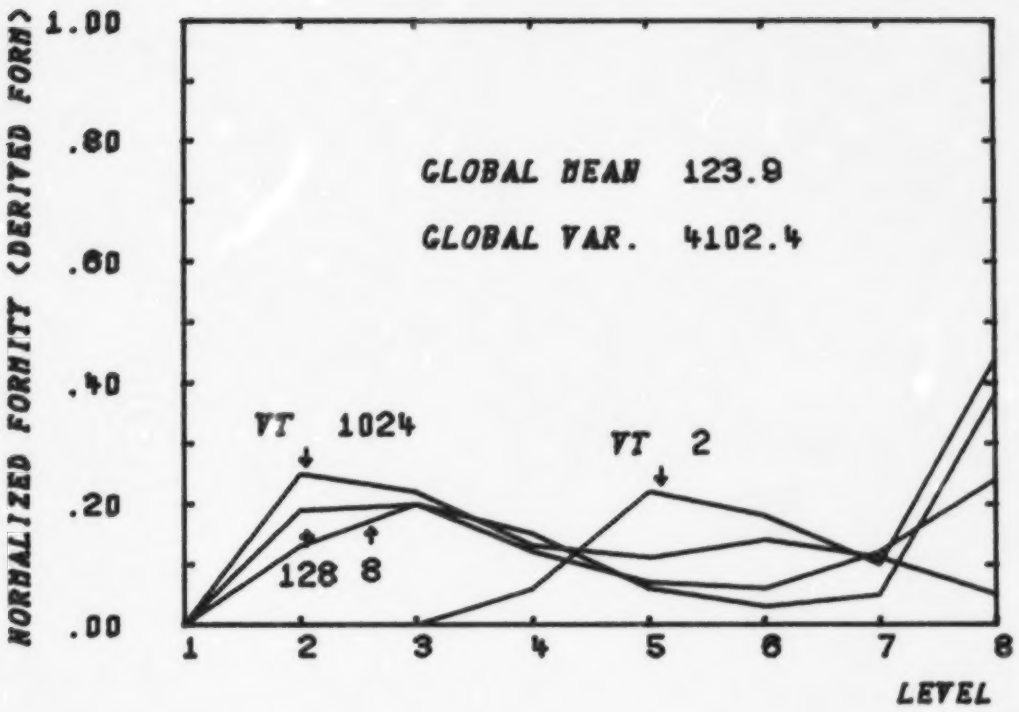


FIGURE 7 - Same low formity image as in Fig. 6 (terminal node method)

There is, however, a final comment on the formity curves (Figs. 5, 6, 7) that should be made concerning the representation of the formity itself. In the introduction, we stated that one of our main aims was the objectivity of the analysis process. Also, a small amount of output data requiring no human interpretation was another of our important goals. Since the formity curves did not fully meet this requirement, an additional effort was made to summarize the formity results more compactly, which led to the 'total' formity indicator.

3.3 The Total Formity Indicator

Two different forms of the total formity indicator have been retained. The first one is named the total normalized formity (TNFORM) and is simply the total of the normalized formity of each level using the intermediate node method. The second form is called the total weighted formity (TWFORM) and is obtained by summing, for all levels, the product of the actual number of subimages with a weighting factor of that level.

Analytically, we have

$$TNFORM = \sum_{i=1}^{k} \alpha_i \times 4^i \qquad [3]$$

and

$$TWFORM = \sum_{i=1}^{k} \alpha_i \times \lambda^i \qquad [4]$$

$$\text{with } \lambda_i = 4^{\frac{i}{2}(i-1)} \quad \text{for all } i$$

where

TNFORM = the total normalized formity,
TWFORM = the total weighted formity,

α_i = the number of subimages in level i using the intermediate node method,

λ_i = the weighting factor of level i,

i = the current level excluding level #0 (the global image level),

k = the maximum level of segmentation (i.e. the pixel level).

The minimum and maximum of TNFORM and TWFORM are:

$$0 \leq \text{TNFORM} \leq k \text{ (real number)}$$

$$0 \leq \text{TWFORM} \leq \sum_{i=1}^{k} 4^{\frac{1}{2}\left[i(i+1)\right]} \text{(integer number)}$$

Table IV shows some examples of results obtained using the image in Fig. 1 and with real data from Figs. 5 and 6.

We observe that the TNFORM varies over a very limited range and is thus easily accommodated by a computer. However, the decomposition of the TNFORM indicator does not lead to a unique set of α_i, as shown in Appendix A. On the other hand, the TWFORM indicator is decomposable into a unique set of α_i when only the number of levels of segmentation is known. In other words, it is possible to retrieve from the TWFORM indicator (which is a single number) the number of actual subimages given by the quad tree calculation because of the uniqueness of TWFORM (see Appendix A). However, the consequence of this very interesting feature is the geometrical growth of TWFORM over non-accommodable ranges.

In practical terms, TWFORM cannot actually be used in most situations because of the astronomical numbers involved in its calculation. However, it is interesting to know that it is possible to summarize the formity of an image in a single decomposable number. Other forms of the total formity indicator will be studied in order to find one that can be accommodated and decomposed.

In summary, both forms of the total formity indicator (TNFORM and TWFORM) allow representation of the geometric complexity in a picture by a "single number" (eqs. 3 and 4). The TNFORM indicator is probably the only practical form for most applications, but the TWFORM is also very interesting for some limited cases since it is possible to decompose the indicator to retrieve the number of subimages of each level.

TABLE IV

Typical results

Identification	Max. level of segmentation	TNFORM* obtained	TWFORM* obtained	TWFORM maximum
Picture in Fig. 1	4	2.34375	165,684	1,052,748
High formity image (Fig. 5)	8	7.8257	$3.964... \times 10^{21}$	$4.72... \times 10^{21}$
Low formity image (Fig. 6)	8	5.24260	$1.807... \times 10^{21}$	$4.72... \times 10^{21}$

* The variance threshold was eight for the formity calculation (i.e. VT = 8).

4.0 CONCLUSION

The utilization of new derivatives of the quad tree appears to be very helpful for image analysis. They make it possible to represent the geometric complexity of an image in a very compact form. Moreover, the calculation of the quad tree derived indicators is very interesting from an engineering point of view because it does not require a large-memory computer or an exaggerated number of floating point operations. Once the variance threshold has been fixed, the analysis does not require any active human interpretation.

For all these reasons, the proposed analysis is adequate for large data banks and is also 'hardware implementable' in an electro-optical system.

5.0 ACKNOWLEDGEMENTS

The author wishes to thank the following persons for their direct or indirect support of this work: Messrs. L.S. Davis from the University of Maryland, G. McLaughlin from DREV Armaments Division, A. Blanchard from DREV Data Systems Division and D. Beauchamp from l'Université de Sherbrooke.

6.0 REFERENCES

1. Rosenfeld, A., "Quad Trees and Pyramids: Hierarchical Representation of Images", TR-1171, University of Maryland, College Park, Maryland, May 1982.

2. Rosenfeld, A., Samet, H., Shaffer, C. and Webber, R., "Application of Hierarchical Data Structures to Geographical Information Systems", TR-1197, University of Maryland, College Park, Maryland, June 1982.

3. Miller, I. and Freund, J.E., "Probability and Statistics for Engineers", Prentice-Hall, New Jersey, 1977.

APPENDIX A

Decomposition of Total Formity Indicators

Recalling eqs. 3 and 4 of Section 3.3, we have

$$TNFORM = \sum_{i=1}^{k} \alpha_i \times 4^i$$

$$TWFORM = \sum_{i=1}^{k} \alpha_i \times \lambda_i \text{ with } \lambda_i = 4^{\frac{i}{2}(i-1)} \quad \text{for all } i$$

where TNFORM is the total normalized formity,
TWFORM is the total weighted formity,
i is the current level of segmentation,
k is the maximum level of segmentation (i.e. pixel level),
α_i is the number of subimages of level i (with intermediate node method).

It has been stated that the TWFORM indicator can be decomposed into a unique set of α_i and that the TNFORM indicator cannot be.

To show that these affirmations are correct, we will assume that two different sets of factors (α_i and β_i) give the same total formity, and we will try to prove that the set of β_i must be equal to that of α_i for TWFORM but not necessarily for TNFORM.

First, let us consider the case of TNFORM. Analytically, we have

$$\text{TNFORM} = \sum_{i=1}^{k} \alpha_i/4^i$$

$$\text{TNFORM} = \sum_{i=1}^{k} \beta_i/4^i$$

so $$\sum_{i=1}^{k} \frac{\alpha_i}{4^i} = \sum_{i=1}^{k} \frac{\beta_i}{4^i}$$

$$\sum_{i=1}^{k} \frac{\alpha_i}{4^i} - \sum_{i=1}^{k} \frac{\beta_i}{4^i} = 0$$

$$\sum_{i=1}^{k} \left(\frac{\alpha_i - \beta_i}{4^i}\right) = 0$$

$$\frac{(\alpha_1-\beta_1)}{4} + \frac{(\alpha_2-\beta_2)}{4^2} + \frac{(\alpha_3-\beta_3)}{4^3} + \ldots \frac{(\alpha_k-\beta_k)}{4^k} = 0$$

An obvious solution of this equation is

$\alpha_i = \beta_i$ for all i

but this is not the sole solution. Another possible solution appears when two (or more) terms cancel each other out. This eventuality is quite possible since all $\frac{\alpha_i - \beta_i}{4^i}$ terms are of the same order of magnitude varying between zero and one. It can be easily verified using a simple hypothetical case that two different sets may lead to the same TNFORM indicator, as shown in Table A-I.

TABLE A-I

Level i	Maximum number of subimages	Picture in Fig. 1		Hypothetical possible case	
		α_i	$\alpha_i / 4^i$	α_i	$\alpha_i / 4^i$
1	4	4	1	4	1
2	16	12	.75	12	.75
3	64	28	.4375	24	.375
4	256	40	.15625	56	.21875

TNFORM: 2.34375 TNFORM: 2.34375

For TWFORM, the development is very similar:

$$TWFORM = \sum_{i=1}^{k} \alpha_i \times \lambda_i$$

$$\text{TWFORM} = \sum_{i=1}^{k} \beta_i \times \lambda_i$$

$$\sum_{i=1}^{k} \alpha_i \times \lambda_i = \sum_{i=1}^{k} \beta_i \times \lambda_i$$

$$\sum_{i=1}^{k} \alpha_i \times \lambda_i = \sum_{i=1}^{k} \beta_i \times \lambda_i$$

$$\sum_{i=1}^{k} (\alpha_i - \beta_i) \times \lambda_i = 0$$

$$(\alpha_1-\beta_1) \times 1 + [(\alpha_2 - \beta_2) \times 4] + (\alpha_3-\beta_3) \times 4^3 + \ldots + (\alpha_k-\beta_k) \times 4^{\frac{k}{2}(k-1)} = 0$$

[5]

It may be expected that two solutions also exist for this equation, but in this case the second solution can be rejected because of the quadratic property of the quad tree and the geometrical growth of λ_i. In fact the weighting factors λ_i have been chosen to insure that each of the $\alpha_i \times \lambda_i$ terms has a range that does not overlap with another term. So, according to [5], no $(\alpha_i - \beta_i) \times \lambda_i$ term may cancel out another term (or a combination of other terms), and thus all terms have to be zero to satisfy eq. 5. Automatically, all $\alpha_i - \beta_i$ terms have to be zero which proves that a given TWFORM is composed of only one combination of α_i.

To illustrate more formally the uniqueness of the α_i for a given TWFORM, the minimum and the maximum of all terms of eq. 5 are summarized in Table A-II.

TABLE A-II

i	λ_i	$\alpha_i\lambda_i$ minimum	$\alpha_i\lambda_i$ maximum	$\alpha_i\lambda_i$ minimum step
1	1	0	4	1
2	4	0	4^3	4
3	4^3	0	4^6	4^3
4	4^6	0	4^{10}	4^6
5	4^{10}	0	4^{15}	4^{10}
.	.	.	.	
.	.	.	.	
.	.	.	.	
k	$4^{\frac{k}{2}(k-1)}$	0	$4^{\frac{k}{2}(k-1)}$	$4^{\frac{k}{2}(k-1)}$

Note that, with selected weighting, the minimum step of product $\alpha_i\lambda_i$ is as big as the maximum product of the previous level. This insures that no term of the summation would vary into the range of another level.

The main advantage of this weighting factor is that it makes the decomposition process very easy. The TWFORM is simply divided by the weighting factor of the last level to get the first estimation of the α_i of this level. Using some quad tree properties, the exact α_i of the level can be found and the decomposition may go on to the next highest level.

CRDV R-4318/84 (NON CLASSIFIE)

Bureau - Recherche et Développement, MDN, Canada.
CRDV, C.P. 8800, Courcelette, Qué. G0A 1R0

"Un indicateur de type 'quad tree' pour l'analyse et la classification des arrières-plans infrarouges"
par R. Charpentier

Un nouvel indicateur géométrique a été conçu et mis au point pour permettre l'analyse et la classification de grandes banques d'images infrarouges. Il permet d'estimer la 'non-uniformité' dans une image de façon objective et relativement simple et d'en exprimer le résultat sous forme très compacte. Il a été nommé indicateur de 'formité' et est dérivé de la théorie de la segmentation de type "quad tree". Quelques résultats obtenus avec des images infrarouges sont présentés pour illustrer les propriétés de cet indicateur.

CRDV R-4318/84 (NON CLASSIFIE)

Bureau - Recherche et Développement, MDN, Canada.
CRDV, C.P. 8800, Courcelette, Qué. G0A 1R0

"Un indicateur de type 'quad tree' pour l'analyse et la classification des arrières-plans infrarouges"
par R. Charpentier

Un nouvel indicateur géométrique a été conçu et mis au point pour permettre l'analyse et la classification de grandes banques d'images infrarouges. Il permet d'estimer la 'non-uniformité' dans une image de façon objective et relativement simple et d'en exprimer le résultat sous forme très compacte. Il a été nommé indicateur de 'formité' et est dérivé de la théorie de la segmentation de type "quad tree". Quelques résultats obtenus avec des images infrarouges sont présentés pour illustrer les propriétés de cet indicateur.

CRDV R-4318/84 (NON CLASSIFIE)

Bureau - Recherche et Développement, MDN, Canada.
CRDV, C.P. 8800, Courcelette, Qué. G0A 1R0

"Un indicateur de type 'quad tree' pour l'analyse et la classification des arrières-plans infrarouges"
par R. Charpentier

Un nouvel indicateur géométrique a été conçu et mis au point pour permettre l'analyse et la classification de grandes banques d'images infrarouges. Il permet d'estimer la 'non-uniformité' dans une image de façon objective et relativement simple et d'en exprimer le résultat sous forme très compacte. Il a été nommé indicateur de 'formité' et est dérivé de la théorie de la segmentation de type "quad tree". Quelques résultats obtenus avec des images infrarouges sont présentés pour illustrer les propriétés de cet indicateur.

CRDV R-4318/84 (NON CLASSIFIE)

Bureau - Recherche et Développement, MDN, Canada.
CRDV, C.P. 8800, Courcelette, Qué. G0A 1R0

"Un indicateur de type 'quad tree' pour l'analyse et la classification des arrières-plans infrarouges"
par R. Charpentier

Un nouvel indicateur géométrique a été conçu et mis au point pour permettre l'analyse et la classification de grandes banques d'images infrarouges. Il permet d'estimer la 'non-uniformité' dans une image de façon objective et relativement simple et d'en exprimer le résultat sous forme très compacte. Il a été nommé indicateur de 'formité' et est dérivé de la théorie de la segmentation de type "quad tree". Quelques résultats obtenus avec des images infrarouges sont présentés pour illustrer les propriétés de cet indicateur.

DREV R-4318/84 (UNCLASSIFIED)

Research and Development Branch, DND, Canada.
DREV, P.O. Box 8800, Courcelette, Que. G0A 1R0

"A Quad Tree Derived Indicator for Infrared Background Analysis and Classification"
by R. Charpentier

A new geometric indicator is proposed for analyzing and classifying infrared images. The indicator, which was named 'formity', provides an objective estimate of the 'nonuniformity' of an image in a very compact form. Derived from the quad tree theory, this indicator is relatively simple to apply and should be particularly useful in relation to large banks of images. Some typical results obtained from thermal images are presented to illustrate its properties.

DREV R-4318/84 (UNCLASSIFIED)

Research and Development Branch, DND, Canada.
DREV, P.O. Box 8800, Courcelette, Que. G0A 1R0

"A Quad Tree Derived Indicator for Infrared Background Analysis and Classification"
by R. Charpentier

A new geometric indicator is proposed for analyzing and classifying infrared images. The indicator, which was named 'formity', provides an objective estimate of the 'nonuniformity' of an image in a very compact form. Derived from the quad tree theory, this indicator is relatively simple to apply and should be particularly useful in relation to large banks of images. Some typical results obtained from thermal images are presented to illustrate its properties.

DREV R-4318/84 (UNCLASSIFIED)

Research and Development Branch, DND, Canada.
DREV, P.O. Box 8800, Courcelette, Que. G0A 1R0

"A Quad Tree Derived Indicator for Infrared Background Analysis and Classification"
by R. Charpentier

A new geometric indicator is proposed for analyzing and classifying infrared images. The indicator, which was named 'formity', provides an objective estimate of the 'nonuniformity' of an image in a very compact form. Derived from the quad tree theory, this indicator is relatively simple to apply and should be particularly useful in relation to large banks of images. Some typical results obtained from thermal images are presented to illustrate its properties.

DREV R-4318/84 (UNCLASSIFIED)

Research and Development Branch, DND, Canada.
DREV, P.O. Box 8800, Courcelette, Que. G0A 1R0

"A Quad Tree Derived Indicator for Infrared Background Analysis and Classification"
by R. Charpentier

A new geometric indicator is proposed for analyzing and classifying infrared images. The indicator, which was named 'formity', provides an objective estimate of the 'nonuniformity' of an image in a very compact form. Derived from the quad tree theory, this indicator is relatively simple to apply and should be particularly useful in relation to large banks of images. Some typical results obtained from thermal images are presented to illustrate its properties.